

install.packages("cfbfastR")
CRAN
remotes::install_github("sportsdataverse/cfbfastR")
library(cfbfastR)

234 exports in five families: load_cfb_* /
load_espn_cfb_* / load_ncaa_mfb_* (release data),
cfbd_* (CollegeFootballData API, key required),
espn_cfb_*, fox_cfb_*, yahoo_cfb_* (no key), plus
PBP / EPA helpers. Every function returns a tibble with a
cfbfastR_type attribute.

CFBD API KEY

All cfbfd_* calls need a free key from
collegefootballdata.com, stored as CFBD_API_KEY. Persist it
in .Renviron:

usethis::edit_r_environ()
then add the line (no quotes), save, restart
R:
CFBD_API_KEY = YOUR-API-KEY-HERE

Or for the current session only (with quotes):

Sys.setenv(CFBD_API_KEY = "YOUR-API-KEY-HERE")
cfbd_key() # read it back
has_cfbd_key() # TRUE / FALSE
cfbd_api_key_info() # remaining calls /
rate limit

?register_cfbd documents the idiom; requests fail fast
with a pointer there when the key is missing.

INCLUDED DATA

Object	Contents
cfbd_play_ty pe_df	45 play types: id, text, abbreviation (use in play_type=)
cfbd_conf_ty pes_df	11 conferences: id, name, short_name, abbreviation

Docs: ?data. Pass the abbreviation (e.g. "SEC", "BIG") to
any conference= argument.

load_cfb_pbp() reads the pre-built, EPA/WPA-enriched
play-by-play .rds files published to the sportsdataverse-
data GitHub release cfbfastR_cfb_pbp (game manifest
from cfbfastR-data). Seasons 2014 to present; no API key.

bpb <- load_cfb_pbp(seasons = 2024)
pbp <- load_cfb_pbp(seasons = 2021:2024)
bpb <- load_cfb_pbp(seasons = TRUE) # all
2014+

CFBD-backed season loaders (need
CFBD_API_KEY)
sched <- load_cfb_schedules(seasons = 2024)
2001+
roster <- load_cfb_rosters(seasons = 2024)
2001+
teams <- load_cfb_teams(fbs_only = TRUE)

Loader	Wraps / source
load_cfb_pbp	release cfbfastR_cfb_pbp, 1 file per season
load_cfb_schedules	cfbd_game_in fo() per season
load_cfb_rosters	cfbd_team_ro ster() per season
load_cfb_teams	cfbd_team_in fo()
load_espn_cfb_* (28) · load_ncaa_mfb_* (13)	ESPN + stats.ncaa.org release datasets (3.0)
load_cfb_ratings(weekly) fpi_weekly team_talent recruits returning_production *_crosswalk	ratings_recruiting · crosswalk releases (3.0)

Progress bars appear automatically when progressr is
installed.

LOCAL PBP DATABASE

build / refresh a SQLite db of every season
update_cfb_db(dbdir = ".", dbname =
"cfb_pbp_db",

tblname = "cfbfastR_pbp",
force_rebuild = FALSE) # TRUE =
rebuild all
or hand in your own DBI connection
con <- DBI::dbConnect(RSQLite::SQLite(),
"cfb_pbp_db")
update_cfb_db(db_connection = con)
load_cfb_pbp can write straight into it
load_cfb_pbp(2024, dbConnection = con,
tablename = "cfbfastR_pbp")

Default directory comes from
getOption("cfbfastR_dbdir"). Needs DBI +
RSQLite (Suggests).

espn_cfb_scoreboard(date = 20241130) #
YYYYMMDD
espn_cfb_schedule(year = 2024, week = 1,
season_type = "regular",

groups = 80)
espn_cfb_calendar(year = 2024)
espn_cfb_teams()
espn_cfb_team_roster(team_id = 52, year =
2024)
espn_cfb_team_stats(team_id = 52, year = 2024)
espn_cfb_player_stats(athlete_id = 4426502,
year = 2024)
espn_cfb_rankings(year = 2024)
espn_cfb_standings(year = 2024, group_id = 90)

Play-by-play & game detail

espn_cfb_pbp(game_id = 401628334, epa_wpa =
TRUE)
espn_cfb_pbp_v2(401628334, epa_wpa = TRUE,
output = "default") #

lean | full
espn_cfb_game_pbp(401628334, participants =
"wide")
espn_cfb_game_drives(401628334, plays =
"expand")
espn_cfb_game_player_box(401628334)
espn_cfb_game_odds(401628334)
espn_cfb_game_predictor(401628334)

Ratings & metrics

espn_ratings_fpi(year = 2024) # FPI
table
espn_cfb_powerindex(year = 2024);
espn_cfb_qbr(year = 2024)
espn_metrics_wp(game_id = 401628334) # ESPN
win prob
espn_cfb_recruits(year = 2025, max_results =
300)
espn_cfb_clear_cache() # drop
memoised ESPN

More: espn_cfb_game_* · espn_cfb_team_* ·
espn_cfb_player_* families.

FOX & YAHOO (NO KEY)

fox_cfb_pbp(game_id);
fox_cfb_boxscore(game_id)
fox_cfb_odds(game_id);
fox_cfb_team_roster(team_id)
fox_cfb_standings(team_id);
fox_cfb_team_gamelog(team_id)
fox_cfb_league_leaders(category = "passing",
who = "player")
yahoo_cfb_scoreboard(season = 2024, week = 1)
yahoo_cfb_boxscore(game_id)
yahoo_cfb_player_season_stats(season = 2024)
yahoo_cfb_team_season_stats(season = 2024)

Famil y
Game s
Drives /
plays
Team s
Playe rs /
stats
Recru iting
Ratin gs
Bettin g
Metric s
Draft /
schoo ls
Live

Functions
cfbd_game_info _box_advanced _team_stats
_player_stats _records _media _weather
cfbd_calendar
cfbd_drives cfbfd_plays cfbfd_play_types
cfbd_play_stats_* cfbfd_pbp_data
cfbd_team_info _roster _talent _matchup
cfbd_teams_fbs
cfbd_conference_affiliations _changes
cfbd_player_info _usage _returning
cfbd_stats_season_* _game_*
cfbd_stats_player_success*
cfbd_player_season_overview
cfbd_recruiting_player _team _position
_transfer_portal
cfbd_ratings_sp _srs _srs_expanded _elo
_fpi_core cfbfd_rankings
cfbd_playoffs_cfp*
cfbd_betting_lines cfbfd_betting_ats
cfbd_metrics_ppa_* _wp _wp_pregame
_fg_ep _wepa_*
cfbd_draft_* cfbfd_coaches _profile
_seasons _tenures cfbfd_venues
cfbd_conferences
cfbd_live_scoreboard cfbfd_live_plays

PYTHON MIRROR

pip install sportsdataverse
from sportsdataverse.cfb import (load_cfb_pbp,
load_cfb_schedule, load_cfb_rosters,
CFBDPlayProcess, espn_cfb_schedule)
bpb = load_cfb_pbp(seasons=[2024]) #
polars
proc = CFBDPlayProcess(gameId=401628334)
df = proc.run_processing_pipeline() #
EPA/WPA

sportsdataverse.cfb adds load_cfb_team_box ,
load_cfb_player_box , load_cfb_drives ,
load_cfb_ratings , load_cfb_recruits , and more; all
from the same cfbfastR-data releases.
py.sportsdataverse.org

SPORTSDATVERSE SIBLINGS

recruitR	recruiting rankings & commits
cfbplotR	team logos / colors for ggplot2
cfb4th	fourth-down decision model
cfbseedR	conference / playoff seeding sims
nflfastR	same EPA / WP design for the NFL

cfbfastR 2.3.0 · Updated 2026-08 · page 1 / 2

CC BY SA · SportsDataverse · cfbfastR.sportsdataverse.org · source: github.com/sportsdataverse/cfbfastR



GAMES · DRIVES · PLAYS

GAMES DATA

```
library(cfbfastR)
cfbd_game_info(2024, week = 1, team = "Georgia",
               quarter_scores = TRUE)
cfbd_game_team_stats(2024, week = 1, rows_per_team = 1)
cfbd_game_player_stats(2024, week = 1, category = "passing")
cfbd_game_box_advanced(game_id = 401628334, long = FALSE)
cfbd_game_records(2024); cfbd_game_media(2024, week = 1)
cfbd_game_weather(2024, 1)
cfbd_calendar(2024) # week
start/end dates
```

season_type: regular / postseason / both · division: fbs (default) / fcs / ii / iii.

DRIVES DATA

```
cfbd_drives(2024, week = 1, team = "Texas")
cfbd_drives(2024, offense_team = "Oregon", defense_conference = "B1G")
```

PLAYS DATA

```
cfbd_plays(2024, week = 1, team = "Ohio State")
cfbd_plays(2024, week = 3, play_type = 8) # see play_type_df
cfbd_play_types() # id / text / abbr
cfbd_play_stats_types()
cfbd_play_stats_player(2024, week = 1, team = "Alabama")
cfbd_live_scoreboard(division = "fbs")
cfbd_live_plays(game_id = 401628334) # in-progress game
```

CFBD_PLAYS VS CFBD_PBP_DATA

	cfbd_plays()	cfbd_pbp_data()
Source	raw CFBD plays endpoint	plays + drives + games, cleaned
Model ed	no	epa_wpa = TRUE adds EP / EPA / WP / WPA
Parsin g	none	play text, yardage, penalties, players
Scope	any filter (team, conf, type)	one week (or team) at a time
<pre>cfbd_pbp_data(2024, week = 5, epa_wpa = TRUE) cfbd_pbp_data_v2(2024, week = 5, epa_wpa = TRUE, output = "lean") # default lean full</pre>		

For whole seasons use `load_cfb_pbp()` instead (page 1); ESPN twins: `espn_cfb_pbp()` / `espn_cfb_pbp_v2()` .

TEAMS · PLAYERS · STATS · RECRUITING

TEAMS DATA

```
cfbd_team_info(conference = "SEC", year = 2024)
cfbd_team_info(only_fbs = FALSE) # all divisions
cfbd_team_roster(2024, team = "Michigan")
cfbd_team_talent(year = 2024) # 247 composite
cfbd_team_matchup("Auburn", "Alabama", min_year = 2000)
cfbd_team_matchup_records("Texas", "Oklahoma")
```

PLAYERS DATA

```
cfbd_player_info("Bo Nix", position = "QB")
cfbd_player_usage(2024, team = "Oregon", excl_garbage_time = TRUE)
cfbd_player_returning(2025, conference = "B12")
```

STATS DATA

```
cfbd_stats_season_team(2024, team = "Notre Dame")
cfbd_stats_season_player(2024, category = "rushing",
                         start_week = 1, end_week = 8)
cfbd_stats_season_advanced(2024, excl_garbage_time = TRUE)
cfbd_stats_game_advanced(2024, week = 2, team = "LSU")
cfbd_stats_game_havoc(2024, team = "Penn State")
cfbd_stats_categories() # valid category=values
```

RECRUITING DATA

```
cfbd_recruiting_player(2025, team = "Texas", recruit_type = "HighSchool")
cfbd_recruiting_player(2025, state = "FL", position = "WR")
cfbd_recruiting_team(2025) # class ranks
cfbd_recruiting_position(2020, 2025, team = "Ohio State")
cfbd_recruiting_transfer_portal(2025)

recruit_type: "HighSchool", "JUCO", "PrepSchool" .
Richer recruiting surface in recruitR.
```

RATINGS · BETTING · METRICS · DRAFT

RANKINGS & RATINGS

```
cfbd_rankings(2024, week = 10) # AP, Coaches, CFP
cfbd_ratings_sp(2024) # SP+ team
cfbd_ratings_sp_conference(2024, conference = "SEC")
cfbd_ratings_srs(2024, team = "Indiana")
cfbd_ratings_elo(2024, week = 8)
cfbd_ratings_fpi(2024) # ESPN FPI via CFBD
```

BETTING DATA

```
cfbd_betting_lines(year = 2024, week = 1, team = "Georgia")
cfbd_betting_lines(game_id = 401628334, line_provider = "consensus")
cfbd_betting_ats(2024, conference = "ACC") # ATS records
```

METRICS DATA (PPA · WP · EPA)

```
cfbd_metrics_ppa_teams(2024, excl_garbage_time = TRUE)
cfbd_metrics_ppa_games(2024, week = 1, team = "Miami")
cfbd_metrics_ppa_players_season(2024, position = "QB",
                                threshold = 100)
cfbd_metrics_ppa_players_games(2024, week = 1)
cfbd_metrics_ppa_predicted(down = 3, distance = 7)
cfbd_metrics_wp(game_id = 401628334) # per-play WP
cfbd_metrics_wp_pregame(2024, week = 1)
cfbd_metrics_fg_ep() # FG expected points
cfbd_metrics_wepa_team_season(2024)
cfbd_metrics_wepa_players_passing(2024, team = "Ole Miss")
cfbd_metrics_wepa_players_rushing(2024)
cfbd_metrics_wepa_players_kicking(2024)
```

NFL DRAFT DATA

```
cfbd_draft_picks(2025, college = "Georgia")
cfbd_draft_picks(year = 2024, nfl_team = "Chicago")
cfbd_draft_teams(); cfbd_draft_positions()
```

SCHOOLS · MODELS · HELPERS

SCHOOLS DATA

```
cfbd_coaches(team = "Georgia", min_year = 2016)
cfbd_coaches(last = "Saban")
cfbd_venues(); cfbd_conferences()
```

EP / WP MODEL FUNCTIONS

The modeled columns in `load_cfb_pbp()` , `cfbd_pbp_data()` and `espn_cfb_pbp()` come from these exported helpers (multinomial EP, FG and naive WP models are internal objects):

Function	Role
<code>create_epa(play_df, ep_model, fg_model)</code>	EP before/after, EPA, FG adjustment
<code>epa_fg_probs(dat, current_probs, ep_model, fg_mod)</code>	blend FG make prob into EP
<code>create_wpa_naive(df, wp_model)</code>	wp_before / wp_after / wpa
<code>wpa_calcs_naive(df)</code>	WPA bookkeeping across plays
<code>prep_epa_df_after(dat)</code>	end-of-play state for EP after

Exported for reproducibility; most users only toggle `epa_wpa = TRUE`. No `add_wp()` / `create_wpa()` exports exist.

PBP PARSING HELPERS

Function	Adds
<code>clean_pbp_dat(play_df)</code>	play-type flags, scoring, turnovers
<code>clean_drive_dat / clean_drive_info</code>	drive results & ids
<code>clean_play_text</code>	normalised play_text
<code>add_yardage</code>	yds_* from play text
<code>add_player_cols</code>	passer / rusher / receiver names
<code>add_play_counts</code>	per-game / per-drive counters
<code>penalty_detection</code>	penalty flags, yards, declined

Called internally by `cfbd_pbp_data()` ; docs: ? `helpers_pbp` .

CONVENTIONS

`year` is the season start year. Team arguments take the CFBD school name ("Ole Miss", "Miami"); conferences take the abbreviation ("SEC", "B1G", "B12", "ACC"). `excl_garbage_time = TRUE` on stats / PPA / usage.

GET STARTED

```
install.packages("cbbfastR") # CRAN
remotes::install_github("sportsdataverse/cbbfastR")
library(cbbfastR)
```

234 exports in five families: `load_cfb_*` / `load_espn_cfb_*` / `load_ncaa_mfb_*` (release data), `cbbfd_*` (CollegeFootballData API, key required), `espn_cfb_*`, `fox_cfb_*`, `yahoo_cfb_*` (no key), plus PBP / EPA helpers. Every function returns a tibble with a `cbbfastR_type` attribute.

CFBD API KEY

All `cbbfd_*` calls need a free key from collegefootballdata.com, stored as `CFBD_API_KEY`. Persist it in `.Renviron`:

```
usethis::edit_r_environ()
# then add the line (no quotes), save, restart R:
CFBD_API_KEY = YOUR-API-KEY-HERE
```

Or for the current session only (with quotes):

```
Sys.setenv(CFBD_API_KEY = "YOUR-API-KEY-HERE")
cbbfd_key() # read it back
has_cbbfd_key() # TRUE / FALSE
cbbfd_api_key_info() # remaining calls / rate limit
```

?`register_cbbfd` documents the idiom; requests fail fast with a pointer there when the key is missing.

INCLUDED DATA

Object	Contents
<code>cbbfd_play_ty</code>	45 play types: id, text, abbreviation
<code>pe_df</code>	(use in <code>play_type=</code>)
<code>cbbfd_conf_ty</code>	11 conferences: id, name, short_name, abbreviation
<code>pes_df</code>	

Docs: ?`data`. Pass the abbreviation (e.g. "SEC", "B1G") to any `conference=` argument.

DATA LOADERS (RELEASE DATA)

`load_cfb_pbp()` reads the pre-built, EPA/WPA-enriched play-by-play .rds files published to the **sportsdataverse-data** GitHub release `cbbfastR_cfb_pbp` (game manifest from **cbbfastR-data**). Seasons 2014 to present; no API key.

```
bpb <- load_cfb_pbp(seasons = 2024)
bpb <- load_cfb_pbp(seasons = 2021:2024)
bpb <- load_cfb_pbp(seasons = TRUE) # all 2014+

# CFBD-backed season loaders (need CFBD_API_KEY)
sched <- load_cfb_schedules(seasons = 2024)
# 2001+
roster <- load_cfb_rosters(seasons = 2024)
# 2001+
teams <- load_cfb_teams(fbs_only = TRUE)
```

Loader	Wraps / source
<code>load_cfb_pbp</code>	release <code>cbbfastR_cfb_pbp</code> , 1 file per season
<code>load_cfb_schedules</code>	<code>cbbfd_game_info()</code> per season
<code>load_cfb_rosters</code>	<code>cbbfd_team_roster()</code> per season
<code>load_cfb_teams</code>	<code>cbbfd_team_info()</code>
<code>load_espn_cfb_*</code> (28) · <code>load_ncaa_mfb_*</code> (13)	ESPN+ stats.ncaa.org release datasets (3.0)
<code>load_cfb_ratings(weekly)</code> <code>fpi_weekly</code> <code>team_talent</code> <code>recruits</code> <code>returning_production_*</code> <code>crosswalk</code>	<code>ratings</code> · <code>recruiting</code> · <code>crosswalk</code> releases (3.0)

Progress bars appear automatically when `progressr` is installed.

LOCAL PBP DATABASE

```
# build / refresh a SQLite db of every season
update_cfb_db(dbdir = ".", dbname = "cfb_pbp_db",
              tblname = "cbbfastR_pbp",
              force_rebuild = FALSE) # TRUE =
rebuild all
# or hand in your own DBI connection
con <- DBI::dbConnect(RSQLite::SQLite(),
"cfb_pbp_db")
update_cfb_db(db_connection = con)
# load_cfb_pbp can write straight into it
load_cfb_pbp(2024, dbConnection = con,
              tablename = "cbbfastR_pbp")
```

Default directory comes from `getOption("cbbfastR.dbdir")`. Needs `DBI` + `RSQLite` (suggested).

ESPN LIVE (NO KEY)

```
espn_cfb_scoreboard(date = 20241130) #
YYYYMMDD
espn_cfb_schedule(year = 2024, week = 1,
                  season_type = "regular",
                  groups = 80)
espn_cfb_calendar(year = 2024)
espn_cfb_teams()
espn_cfb_team_roster(team_id = 52, year = 2024)
espn_cfb_team_stats(team_id = 52, year = 2024)
espn_cfb_player_stats(athlete_id = 4426502,
year = 2024)
espn_cfb_rankings(year = 2024)
espn_cfb_standings(year = 2024, group_id = 90)
```

Play-by-play & game detail

```
espn_cfb_pbp(game_id = 401628334, epa_wpa =
TRUE)
espn_cfb_pbp_v2(401628334, epa_wpa = TRUE,
                output = "default") # lean

| full
espn_cfb_game_pbp(401628334, participants =
"wide")
espn_cfb_game_drives(401628334, plays =
"expand")
espn_cfb_game_player_box(401628334)
espn_cfb_game_odds(401628334)
espn_cfb_game_predictor(401628334)
```

Ratings & metrics

```
espn_ratings_fpi(year = 2024) # FPI
table
espn_cfb_powerindex(year = 2024);
espn_cfb_qbr(year = 2024)
espn_metrics_wp(game_id = 401628334) # ESPN win
prob
espn_cfb_recruits(year = 2025, max_results =
300)
espn_cfb_clear_cache() # drop
memoised ESPN
```

More: `espn_cfb_game_*` · `espn_cfb_team_*` · `espn_cfb_player_*` families.

FOX & YAHOO (NO KEY)

```
fox_cfb_pbp(game_id);
fox_cfb_boxscore(game_id)
fox_cfb_odds(game_id);
fox_cfb_team_roster(team_id)
fox_cfb_standings(team_id);
fox_cfb_team_gamelog(team_id)
fox_cfb_league_leaders(category = "passing",
who = "player")
yahoo_cfb_scoreboard(season = 2024, week = 1)
yahoo_cfb_boxscore(game_id)
yahoo_cfb_player_season_stats(season = 2024)
yahoo_cfb_team_season_stats(season = 2024)
```

CFBD_* QUICK MAP (PAGE 2)

Famil y	Functions
Game s	<code>cbbfd_game_info</code> <code>_box_advanced</code> <code>_team_stats</code> <code>_player_stats</code> <code>_records</code> <code>_media</code> <code>_weather</code> <code>cbbfd_calendar</code>
Drives / plays	<code>cbbfd_drives</code> <code>cbbfd_plays</code> <code>cbbfd_play_types</code> <code>cbbfd_play_stats_*</code> <code>cbbfd_pbp_data</code>
Team s	<code>cbbfd_team_info</code> <code>_roster</code> <code>_talent</code> <code>_matchup</code> <code>cbbfd_teams_fbs</code> <code>cbbfd_conference_affiliations</code> <code>_changes</code>
Player s / stats	<code>cbbfd_player_info</code> <code>_usage</code> <code>_returning</code> <code>cbbfd_stats_season_*</code> <code>_game_*</code> <code>cbbfd_stats_player_success*</code> <code>cbbfd_player_season_overview</code>
Recru iting	<code>cbbfd_recruiting_player</code> <code>_team</code> <code>_position</code> <code>_transfer_portal</code>
Ratin gs	<code>cbbfd_ratings_sp</code> <code>_srs</code> <code>_srs_expanded</code> <code>_elo</code> <code>_fpi</code> <code>_core</code> <code>cbbfd_rankings</code> <code>cbbfd_playoffs_cfp*</code>
Bettin g	<code>cbbfd_betting_lines</code> <code>cbbfd_betting_ats</code>
Metric s	<code>cbbfd_metrics_ppa_*</code> <code>_wp</code> <code>_wp_pregame</code> <code>_fg_ep</code> <code>_wepa_*</code>
Draft / schoo ls	<code>cbbfd_draft_*</code> <code>cbbfd_coaches</code> <code>_profile</code> <code>_seasons</code> <code>_tenures</code> <code>cbbfd_venues</code> <code>cbbfd_conferences</code>
Live	<code>cbbfd_live_scoreboard</code> <code>cbbfd_live_plays</code>

PYTHON MIRROR

```
pip install sportsdataverse
from sportsdataverse.cfb import (load_cfb_pbp,
load_cfb_schedule, load_cfb_rosters,
CFBPlayProcess, espn_cfb_schedule)

bpb = load_cfb_pbp(seasons=[2024]) #
polars
proc = CFBPlayProcess(gameId=401628334)
df = proc.run_processing_pipeline() #
EPA/WPA
```

`sportsdataverse.cfb` adds `load_cfb_team_box`, `load_cfb_player_box`, `load_cfb_drives`, `load_cfb_ratings`, `load_cfb_recruits`, and more; all from the same `cbbfastR-data` releases. `py.sportsdataverse.org`

SPORTSDATAVERSE SIBLINGS

<code>recruitR</code>	recruiting rankings & commits
<code>cfbplotR</code>	team logos / colors for ggplot2
<code>cfb4th</code>	fourth-down decision model
<code>cfbseedR</code>	conference / playoff seeding sims
<code>nflfastR</code>	same EPA / WP design for the NFL



CollegeFootballData API :: cfb*_ surface & model helpers

All cfb*_ need CFB*_API_KEY (page 1) · grouped as in the pkgdown reference · Python mirror: [sportsdataverse.cfb](#) · see also [recruitR](#) · [cfb4th](#)

GAMES · DRIVES · PLAYS

GAMES DATA

```
library(cfbfastR)
cfbd_game_info(2024, week = 1, team = "Georgia",
               quarter_scores = TRUE)
cfbd_game_team_stats(2024, week = 1, rows_per_team = 1)
cfbd_game_player_stats(2024, week = 1, category = "passing")
cfbd_game_box_advanced(game_id = 401628334, long = FALSE)
cfbd_game_records(2024); cfbd_game_media(2024, week = 1)
cfbd_game_weather(2024, 1)
cfbd_calendar(2024) # week start/end dates
```

season_type: regular / postseason / both · division: fbs (default) / fcs / ii / iii.

DRIVES DATA

```
cfbd_drives(2024, week = 1, team = "Texas")
cfbd_drives(2024, offense_team = "Oregon", defense_conference = "B1G")
```

PLAYS DATA

```
cfbd_plays(2024, week = 1, team = "Ohio State")
cfbd_plays(2024, week = 3, play_type = 8) # see play_type_df
cfbd_play_types() # id / text / abbr
cfbd_play_stats_types()
cfbd_play_stats_player(2024, week = 1, team = "Alabama")
cfbd_live_scoreboard(division = "fbs")
cfbd_live_plays(game_id = 401628334) # in-progress game
```

CFBD_PLAYS VS CFBD_PBP_DATA

	cfbd_plays()	cfbd_pbp_data()
Source	raw CFBD plays endpoint	plays + drives + games, cleaned endpoint
Model ed	no	epa_wpa = TRUE adds EP / EPA / WP / WPA
Parsin g	none	play text, yardage, penalties, players
Scope	any filter (team, conf, type)	one week (or team) at a time

```
cfbd_pbp_data(2024, week = 5, epa_wpa = TRUE)
cfbd_pbp_data_v2(2024, week = 5, epa_wpa = TRUE,
                 output = "lean") # default
| lean | full
```

For whole seasons use `load_cfb_pbp()` instead (page 1); ESPN twins: `espn_cfb_pbp()` / `espn_cfb_pbp_v2()` .

TEAMS · PLAYERS · STATS · RECRUITING

TEAMS DATA

```
cfbd_team_info(conference = "SEC", year = 2024)
cfbd_team_info(only_fbs = FALSE) # all divisions
cfbd_team_roster(2024, team = "Michigan")
cfbd_team_talent(year = 2024) # 247 composite
cfbd_team_matchup("Auburn", "Alabama", min_year = 2000)
cfbd_team_matchup_records("Texas", "Oklahoma")
```

PLAYERS DATA

```
cfbd_player_info("Bo Nix", position = "QB")
cfbd_player_usage(2024, team = "Oregon", excl_garbage_time = TRUE)
cfbd_player_returning(2025, conference = "B12")
```

STATS DATA

```
cfbd_stats_season_team(2024, team = "Notre Dame")
cfbd_stats_season_player(2024, category = "rushing",
                        start_week = 1, end_week = 8)
cfbd_stats_season_advanced(2024, excl_garbage_time = TRUE)
cfbd_stats_game_advanced(2024, week = 2, team = "LSU")
cfbd_stats_game_havoc(2024, team = "Penn State")
cfbd_stats_categories() # valid category= values
```

RECRUITING DATA

```
cfbd_recruiting_player(2025, team = "Texas", recruit_type = "HighSchool")
cfbd_recruiting_player(2025, state = "FL", position = "WR")
cfbd_recruiting_team(2025) # class ranks
cfbd_recruiting_position(2020, 2025, team = "Ohio State")
cfbd_recruiting_transfer_portal(2025)
```

recruit_type: "HighSchool", "JUCO", "PrepSchool" . Richer recruiting surface in [recruitR](#).

RATINGS · BETTING · METRICS · DRAFT

RANKINGS & RATINGS

```
cfbd_rankings(2024, week = 10) # AP, Coaches, CFP
cfbd_ratings_sp(2024) # SP+ team
cfbd_ratings_sp_conference(2024, conference = "SEC")
cfbd_ratings_srs(2024, team = "Indiana")
cfbd_ratings_elo(2024, week = 8)
cfbd_ratings_fpi(2024) # ESPN FPI via CFBD
```

BETTING DATA

```
cfbd_betting_lines(year = 2024, week = 1, team = "Georgia")
cfbd_betting_lines(game_id = 401628334, line_provider = "consensus")
cfbd_betting_ats(2024, conference = "ACC") # ATS records
```

METRICS DATA (PPA · WP · EPA)

```
cfbd_metrics_ppa_teams(2024, excl_garbage_time = TRUE)
cfbd_metrics_ppa_games(2024, week = 1, team = "Miami")
cfbd_metrics_ppa_players_season(2024, position = "QB",
                                threshold = 100)
cfbd_metrics_ppa_players_games(2024, week = 1)
cfbd_metrics_ppa_predicted(down = 3, distance = 7)
cfbd_metrics_wp(game_id = 401628334) # per-play WP
cfbd_metrics_wp_pregame(2024, week = 1)
cfbd_metrics_fg_ep() # FG expected points
cfbd_metrics_wepa_team_season(2024)
cfbd_metrics_wepa_players_passing(2024, team = "Ole Miss")
cfbd_metrics_wepa_players_rushing(2024)
cfbd_metrics_wepa_players_kicking(2024)
```

NFL DRAFT DATA

```
cfbd_draft_picks(2025, college = "Georgia")
cfbd_draft_picks(year = 2024, nfl_team = "Chicago")
cfbd_draft_teams(); cfbd_draft_positions()
```

SCHOOLS · MODELS · HELPERS

SCHOOLS DATA

```
cfbd_coaches(team = "Georgia", min_year = 2016)
cfbd_coaches(last = "Saban")
cfbd_venues(); cfbd_conferences()
```

EP / WP MODEL FUNCTIONS

The modeled columns in `load_cfb_pbp()` , `cfbd_pbp_data()` and `espn_cfb_pbp()` come from these exported helpers (multinomial EP, FG and naive WP models are internal objects):

Function	Role
<code>create_epa(play_df, ep_model, fg_model)</code>	EP before/after, EPA, FG adjustment
<code>epa_fg_probs(dat, current_probs, ep_model, fg_mod)</code>	blend FG make prob into EP
<code>create_wpa_naive(df, wp_model)</code>	wp_before / wp_after / wpa
<code>wpa_calcs_naive(df)</code>	WPA bookkeeping across plays
<code>prep_epa_df_after(dat)</code>	end-of-play state for EP after

Exported for reproducibility; most users only toggle `epa_wpa = TRUE` .No `add_wp()` / `create_wpa()` exports exist.

PBP PARSING HELPERS

Function	Adds
<code>clean_pbp_dat(play_df)</code>	play-type flags, scoring, turnovers
<code>clean_drive_dat / clean_drive_info</code>	drive results & ids
<code>clean_play_text</code>	normalised play_text
<code>add_yardage</code>	yds_* from play text
<code>add_player_cols</code>	passer / rusher / receiver names
<code>add_play_counts</code>	per-game / per-drive counters
<code>penalty_detection</code>	penalty flags, yards, declined

Called internally by `cfbd_pbp_data()` ; docs: ? [helpers_pbp](#) .

CONVENTIONS

year is the season start year. Team arguments take the CFBD school name ("Ole Miss" , "Miami"); conferences take the abbreviation ("SEC" , "B1G" , "B12" , "ACC") . `excl_garbage_time = TRUE` on stats / PPA / usage.